

Universidade Federal de Santa Catarina
EEL5105: Circuitos e Técnicas Digitais
Semestre: 2024/2 – Projeto

Jogo detector de Código

O projeto final consiste na implementação de um circuito na placa de desenvolvimento *DE2* fazendo uso das estruturas e conhecimentos obtidos durante o curso. O circuito vai implementar a um jogo interativo para a obtenção dum código secreto de 10 bits. O comportamento do jogo está definido a seguir:

- O usuário começa no estado *Init* e da início ao jogo pressionando o botão de pressão *enter* (KEY1). Uma vez no estado *Setup* o usuário deve escolher uma das 16 sequências possíveis com os *Switches* 3 a 0, *SW*(3..0), as quais estão descritas numa memória. Dita memória possui 16 linhas de informação de 10-bits ($2^4 \times 10$) e o aluno pode encontrar um exemplo de memória parcialmente descrita em VHDL no *Moodle* da disciplina. Corre a cargo do aluno preencher a memória com uns códigos à sua escolha. Cada vetor de 10 bits pode ter unicamente quatro "1" lógicos. Os *Switches* 7 a 4, *SW*(7..4) servem para escolher o tempo de jogo por rodada, o aluno pode escolher entre 5 segundos, mínimo e 10 segundos, máximo. Em este estado, os displays HEX5 e HEX3 mostrarão a letra *t* de *tempo* e *n* de *nível* e os displays HEX4 e HEX2 mostrarão o valor do tempo e a linha da memória escolhida.
- Uma vez pressionado *enter* de novo o jogo passa ao estado *Play* e se inicia o jogo. No estado *Play* o usuário tem o valor do tempo escolhido para seleccionar uma sequência usando os *Switches* *SW*(9..0). Em este estado, os displays HEX5 e HEX4 mostrarão a letra *t* de *tempo* e uma contagem ascendente com frequência de 1Hz, respectivamente. É importante destacar o jogador unicamente pode introduzir quatro "1" lógicos por rodada. Se o jogador não pressiona *enter* antes do final da contagem um sinal de status chamado *end_time* é ativado e o jogo vai para o estado *Result*, pelo contrario, se o jogador selecciona uma sequência e pressiona *enter* antes do final da contagem o jogo vai para o estado *Count_Round*.
- No estado *Count_Round* é feito a contagem de rodada e o jogo vai para o estado *Check*. O jogador possui 10 rodadas para adivinhar a sequência.
- No estado *Check* onde será avaliado se:
 - 1) O jogador não introduziu quatro "1" lógicos nos *Switches* onde é ativado um sinal de *status* chamado *sw_erro*.
 - 2) O jogador acabou o numero de rodadas máximo, onde caso tinha chegado a 10 rodadas, um sinal de *status* chamado *end_round* será ativado.
 - 3) O jogador adivinhou as posições dos quatro "1" lógicos da sequência, onde um sinal de *status* chamado *end_game* será ativado.Caso um dos três sinais de *status* esteja ativo, o jogo vai para um estado *Result*, em caso contrario vai para um estado *Wait*.
- No estado *Wait*, os displays HEX3 e HEX2 mostrarão a letra *r* de *round* e a contagem da rodada, respectivamente. Nesse estado será zerada a contagem de tempo. Os displays HEX1 e HEX0 mostrarão a letra *A* de *acertos* e o número de acertos, quando o jogador pressiona *enter* e passa de novo ao estado *Play* para a seguinte rodada.
- No estado *Result* será mostrado nos LEDs vermelhos, *LEDR*(9..0) a sequência a ser adivinhada. No estado *Result* é também mostrado o resultado do numero de pontos em Hexadecimal nos *displays* HEX1 e HEX0. O resultado final seguirá a seguinte formula $16 \times \overline{end_game} + \overline{Round}$, onde *Round* é o número de rodadas (a fórmula está expressada em formato vetor no *datapath*). O usuário deverá pressionar *enter* para passar ao estado *Init* e iniciar outra rodada.
- Um usuário pode em qualquer momento parar o jogo usando o botão de pressão *reset* (KEY0) zerando o sistema, para assim re-iniciar de novo.
- Visando evitar problemas de temporização em função do aperto de um KEY por um ser humano durar muitos ciclos de *clock*, o *Button Press Synchronizer* (ButtonSync) será fornecido em conjunto com o projeto deve ser utilizado. O *ButtonSync* converte apertos das KEYS em pulsos com período de um ciclo de *clock*.
- O projeto deverá ser implementado **obrigatoriamente** usando a abordagem *datapath-control* vista nas aulas de teoria. Corre ao cargo do aluno projetar o *datapath* e juntar com o controlador.

Orientações Gerais:

- Os testes do projeto poderão ser feitos durante as semanas que antecedem o prazo final usando as ferramentas usadas no semestre. O professor estará disponível para solucionar dúvidas nos horários indicados como *Projeto* no cronograma. Na apresentação de funcionamento do projeto no kit DE1 ou DE2, todos os membros do grupo (**máximo 2 integrantes**) deverão estar presentes: